

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problems Mailbox.**



별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto  
is a true copy from the records of the Korean Intellectual  
Property Office.

출원번호 : 특허출원 2001년 제 19883 호  
Application Number PATENT-2001-0019883

출원년월일 : 2001년 04월 13일  
Date of Application APR 13, 2001

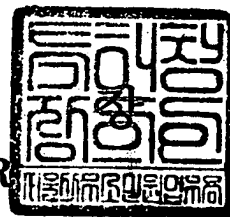
출원인 : 엘지전자주식회사  
Applicant(s) LG ELECTRONICS INC.



2002      년    01      월    14      일

특      허      청

COMMISSIONER



## 【서지사항】

【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0003
【제출일자】	2001.04.13
【국제특허분류】	G06F 013/14
【발명의 명칭】	데이터의 천이 상태에 따른 디버깅 장치 및 방법
【발명의 영문명칭】	DEBUGGING DEVICE AND METHOD BY DATA CHANGING
【출원인】	
【명칭】	엘지전자 주식회사
【출원인코드】	1-1998-000275-8
【대리인】	
【성명】	박장원
【대리인코드】	9-1998-000202-3
【포괄위임등록번호】	2000-027763-7
【발명자】	
【성명의 국문표기】	옥재철
【성명의 영문표기】	OK, Jae Chul
【주민등록번호】	730202-1121118
【우편번호】	137-132
【주소】	서울특별시 서초구 양재2동 204-15 다정주택 102호
【국적】	KR
【취지】	특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대리인 박장원 (인)
【수수료】	
【기본출원료】	20 면 29,000 원
【가산출원료】	0 면 0 원
【우선권주장료】	0 건 0 원
【심사청구료】	0 항 0 원
【합계】	29,000 원
【첨부서류】	1. 요약서·명세서(도면)_1통

**【요약서】****【요약】**

본 발명은 데이터의 천이 상태에 따른 디버깅 장치 및 방법에 관한 것으로, 프로세서 디버깅 중 데이터 메모리의 특정 어드레스를 감시하여 데이터의 변환을 관측하고 그에 상관관계가 있는 어드레스 및 데이터를 출력하여 프로세서를 디버깅 하도록 함을 특징으로 한다. 이를 위하여 본 발명은 프로그램 메모리에 저장된 프로그램에 따라 동작하는 프로세서 코어와; 디버깅을 수행하는 호스트 컴퓨터로부터 명령을 받아 상기 프로세서 코어를 제어하는 디버거 제어부와; 상기 디버거 제어부를 통하여 상기 호스트 컴퓨터로부터 브레이크 포인트 어드레스를 받아 상기 프로세서 코어가 사용하는 프로그램 메모리 어드레스를 관측하고, 동일한 어드레스가 감지되면 디버거 브레이크 포인트로 인식하여 상기 디버거 제어부로 신호를 전달하는 브레이크 포인트 감지부와; 상기 프로세서 코어의 데이터를 저장하는 데이터 메모리와; 상기 데이터 메모리의 어드레스와 데이터를 감시하고 해당 어드레스를 감지시 상기 디버거 제어부로 신호를 보내어 프로세서 코어의 동작을 중지시키고 해당 어드레스와 데이터를 출력하는 메모리 브레이크 컨트롤러로 구성 한다.

**【대표도】**

도 2

## 【명세서】

## 【발명의 명칭】

데이터의 천이 상태에 따른 디버깅 장치 및 방법{DEBUGGING DEVICE AND METHOD BY DATA CHANGING}

## 【도면의 간단한 설명】

도 1은 종래의 디버깅 장치 블록도.

도 2는 본 발명의 데이터의 천이 상태에 따른 디버깅 장치 블록도.

도 3은 본 발명의 메모리 브레이크 컨트롤러 블록도.

도 4는 본 발명의 데이터의 천이 상태에 따른 디버깅 동작 순서도.

\*\*\*\*\* 도면의 주요 부분에 대한 부호의 설명 \*\*\*\*\*

10,20 : 호스트 컴퓨터    11,21 : 디버거 제어부

12,22 : 브레이크 포인트 감지부    13,23 : 프로세서 코어

14,24 : 프로그램 메모리    15,25 : 데이터 메모리

26 : 메모리 브레이크 컨트롤러    31 : 메모리 컨트롤 레지스터, MBCR

32 : 어드레스 레지스터, AR    33 : 어드레스 비교부, AC

34 : 데이터 레지스터, DR    35 : 데이터 비교부, DC

**【발명의 상세한 설명】****【발명의 목적】****【발명이 속하는 기술분야 및 그 분야의 종래기술】**

- <12>        본 발명은 디버깅 장치 및 방법에 관한 것으로, 특히 프로세서 디버깅 중 데이터 흐름을 관측하여 프로세서를 중지 시킬수 있도록 한 데이터 천이 상태에 따른 디버깅 장치 및 방법에 관한 것이다.
- <13>        도 1은 종래의 디버깅 장치에 관한 블록도로서, 이에 도시된 바와 같이 디버깅을 수행하는 호스트 컴퓨터(10)로부터 명령을 받는 디버거 제어부(11)와, 상기 디버거 제어부(11)의 제어를 받아 동작하는 프로세서 코어(13)와, 상기 디버거 제어부(11)로부터 브레이크 포인트 어드레스를 받아 프로세서 코어(13)에서 실행중인 프로그램 메모리(14) 어드레스를 관측하여 동일한 어드레스가 감지되면 디버거 브레이크 포인트로 인식하여 디버거 제어부(11)로 신호를 전달하는 브레이크 포인트 감지부(12)와, 상기 프로세서 코어(13)의 프로그램을 저장하는 프로그램 메모리(14)와; 상기 프로세서 코어(13)의 데이터를 저장하는 데이터 메모리(15)로 구성 된다.
- <14>        이와 같이 구성된 종래의 디버깅 장치의 동작을 설명한다.
- <15>        디버깅 모드로 전환되면, 디버깅을 하기위해 상기 호스트 컴퓨터(10)는 상기 디버거 제어부(11)를 통하여 상기 프로세서 코어(13)를 멈추게 하고, 상기 프로그램 메모리(14) 내에서 프로세서 코어(13)가 멈추기 원하는 프로그램 어드레

스를 상기 브레이크 포인트 감지부(12)에 저장한 후, 상기 프로세서 코어(13)를 상기 프로그램 메모리(14)에 저장된 프로그램 순서대로 동작시킨다.

- <16>       상기 브레이크 포인트 감지부(12)에서는 프로그램 메모리(14)에 저장된 순서대로 동작하는 프로세서 코어(13)를 감시하게 되고, 상기 프로세서 코어(13)가 상기 브레이크 포인트 감지부(12)에 저장된 프로그램 어드레스와 동일한 프로그램 메모리(12)의 프로그램 어드레스를 읽을 경우 상기 브레이크 포인트 감지부(12)는 상기 디버거 제어부(11)로 브레이크 신호를 전달하게 된다. 상기 디버거 제어부(11)는 브레이크 포인트 감지부(12)에서 받은 신호에 의해 프로세서 코어(13)를 멈추게 한 후 호스트 컴퓨터(10)로 제어권을 이동시키고 상기 호스트 컴퓨터(10)는 디버깅 작업을 수행한다.

- <17>       따라서 종래에는, 순차적인 프로그램에서 프로그램 메모리의 특정 어드레스를 브레이크 포인트로 설정하고 프로세서 동작중 이와 상응하는 프로그램 메모리의 어드레스를 패치(Fetch)하는 경우 브레이크 포인트로 인식하여 프로세서의 동작을 중단하는 방법으로 디버깅 하고 있다.

#### 【발명이 이루고자 하는 기술적 과제】

- <18>       그러나, 상기와 같은 종래 기술에 있어서, 데이터 메모리의 관점에서 데이터의 흐름을 파악하기는 불가능하며, 실제 데이터의 흐름 자체가 프로그래밍의 중요한 축을 이루기 때문에 프로그램을 짜는 경우 메모리 할당 오류를 유발시키고 아울러 연산 과정에서 오류 값을 가지게 되어 시스템이 오동작 하는 문제점이 있다.

<19> 따라서, 상기와 같은 문제점을 감안하여 창안한 본 발명은, 프로세서 디버깅 중 데이터 메모리의 특정 어드레스를 감시하여 데이터의 변환을 관측하고 그에 상관관계가 있는 어드레스 및 데이터를 출력하여 프로세서를 디버깅 할수 있는 데이터 천이 상태에 따른 디버깅 장치 및 방법을 제공함에 그 목적이 있다.

**【발명의 구성 및 작용】**

<20> 상기와 같은 목적을 달성하기 위한 본 발명은, 프로그램 메모리에 저장된 프로그램에 따라 동작하는 프로세서 코어와; 디버깅을 수행하는 호스트 컴퓨터로부터 명령을 받아 상기 프로세서 코어를 제어하는 디버거 제어부와; 상기 디버거 제어부를 통하여 상기 호스트 컴퓨터로부터 브레이크 포인트 어드레스를 받아 상기 프로세서 코어가 사용하는 프로그램 메모리의 어드레스를 관측하고, 동일한 어드레스가 감지되면 디버거 브레이크 포인트로 인식하여 상기 디버거 제어부로 신호를 전달하는 브레이크 포인트 감지부와; 상기 프로세서 코어의 데이터를 저장하는 데이터 메모리와; 상기 데이터 메모리의 어드레스와 데이터를 감시하고 해당 어드레스를 감지시 상기 디버거 제어부로 신호를 보내어 프로세서 코어의 동작을 중지시키고 해당 어드레스와 데이터를 출력하는 메모리 브레이크 컨트롤러로 구성된 것을 특징으로 한다.

<21> 또한, 디버거 모드로 전환시, 디버거 제어부의 브레이크 인에이블 신호에 의해 메모리 브레이크 컨트롤 레지스터를 초기화하고 관측대상 어드레스를 어드레스 레지스터에 저장하는 제1단계와; 현재 프로세서 코어가 사용하는 데이터 메모리 어드레스와 상기 관측대상 어드레스를 어드레스 비교부에서 비교하는 제2단계와; 상기 제2단계의 비교결과, 상기 관측대상 어드레스와 현재 프로세서 코어



가 사용하는 데이터 메모리 어드레스가 동일하면, 프로세서 코어가 데이터 메모리의 데이터를 읽어오는 것인지를 판단하는 제3단계와; 상기 제3단계의 판단결과, 상기 프로세서 코어가 데이터 메모리의 데이터를 읽어오면, 브레이크 메모리 컨트롤 레지스터의 어드레스 트레이스 체크 플래그와 데이터 체크 플래그를 셋하고, 다시 현재 프로세서 코어가 사용하는 데이터 메모리 어드레스를 어드레스 비교부에서 비교하는 제4단계와; 상기 제4단계의 비교결과, 상기 관측대상 어드레스와 현재 프로세서 코어가 사용하는 데이터 메모리 어드레스가 동일하면, 프로세서 코어가 데이터를 기록하는 것인지를 판단하는 제5단계와; 상기 제5단계의 판단결과, 상기 프로세서 코어가 데이터 메모리에 데이터를 기록하면, 프로세서 코어의 동작을 중지시키는 제6단계로 동작 하는 것을 특징으로 한다.

<22> 이하 본 발명에 의한 데이터 천이 상태에 따른 디버깅 장치 및 방법의 작용 및 효과를 첨부한 도2 및 도3을 참조하여 설명한다.

<23> 도 2는 본 발명 데이터 천이 상태에 따른 디버깅 장치 및 방법의 구성을 보인 블록도로서, 이에 도시한 바와 같이 프로그램 메모리(24)에 저장된 프로그램에 따라 동작하는 프로세서 코어(23)와, 디버깅을 수행하는 호스트 컴퓨터(20)로부터 명령을 받아 상기 프로세서 코어(23)를 제어하는 디버거 제어부(21)와, 상기 디버거 제어부(21)를 통하여 상기 호스트 컴퓨터(20)로부터 브레이크 포인트 어드레스를 받아 상기 프로세서 코어(23)가 사용하는 프로그램 메모리(24)의 어드레스를 관측하고, 동일한 어드레스가 감지되면 디버거 브레이크 포인트로 인식하여 상기 디버거 제어부(21)로 신호를 전달하는 브레이크 포인트 감지부(22)와, 상기 프로세서 코어(23)의 데이터를 저장하는 데이터 메모리(25)와, 상기 데이터

메모리(25)의 어드레스와 데이터를 감시하고 관측 대상 어드레스를 감지시 상기 디버거 제어부(21)로 신호를 보내어 프로세서 코어(23)의 동작을 중지시키고 해당 어드레스와 데이터를 상기 디버거 제어부(21)를 통하여 상기 호스트 컴퓨터(20)로 출력하는 메모리 브레이크 컨트롤러(26)로 구성 한다.

<24> 또한, 도 3은 본 발명 메모리 브레이크 컨트롤러(26)의 구성을 보인 블록도로서, 이에 도시한 바와 같이 상기 메모리 브레이크 컨트롤러(26)는 상기 디버거 제어부(21)로부터 신호를 받아 메모리 브레이크 컨트롤러(26)를 활성화 시키고, 상기 디버거 제어부(21)로부터 관측대상 어드레스와 그 데이터를 받는 메모리 브레이크 컨트롤 레지스터(31)와, 상기 메모리 브레이크 컨트롤 레지스터(31)로부터 관측대상 어드레스를 받아 저장하는 어드레스 레지스터(32)와, 상기 데이터 메모리(25)로부터 받은 현재 어드레스와 상기 어드레스 레지스터(32)에 저장된 관측대상 어드레스를 비교하는 어드레스 비교부(33)와, 상기 어드레스 레지스터(32)에 저장되어 있는 관측대상 어드레스의 데이터 값을 저장하는 데이터 레지스터와(34), 상기 데이터 메모리(25)로부터 받은 현재 데이터와 상기 데이터 레지스터(34)에 저장되어있는 관측대상 어드레스의 데이터 값을 비교하는 데이터 비교부 (35)로 구성 한다.

<25> 또한, 상기 메모리 브레이크 컨트롤 레지스터(31)는 상기 메모리 브레이크 컨트롤러(26)를 활성화시키는 메모리 브레이크 인에이블 플래그(MBEF)와, 상기 어드레스 비교부(33)에서 상기 어드레스 레지스터(32)에 저장된 어드레스와 동일한 어드레스를 감지해서 그에 해당하는 데이터의 내용을 알았을때 셋(Set) 되는 데이

터 체크 플래그(DCF)와, 최초 관측대상 어드레스를 지정한 후 프로세싱 과정 중에서 관측대상 어드레스의 내용을 읽어 갔을 경우 셋(Set) 되는 어드레스 트레이스 체크 플래그(ACF)로 구성 된다.

<26> 또한, 도4는 본 발명 데이터 천이 상태에 따른 디버깅 방법에 대한 동작 흐름도로서 이에 도시한 바와 같이 디버거 모드로 전환시, 디버거 제어부(21)의 브레이크 인에이블 신호에 의해 메모리 브레이크 컨트롤 레지스터(31)를 초기화(ST1)하고 관측대상 어드레스를 어드레스 레지스터(32)에 저장(ST2)하는 제1단계와, 현재 프로세서 코어(23)가 사용하는 데이터 메모리(25)의 어드레스와 상기 관측대상 어드레스를 어드레스 비교부(33)에서 비교(ST3)하는 제2단계와, 상기 제2단계의 비교결과, 상기 관측대상 어드레스와 현재 프로세서 코어(23)가 사용하는 데이터 메모리(25)의 어드레스가 동일하면, 프로세서 코어(23)가 데이터 메모리(25)의 데이터를 읽어오는 것인지를 판단(ST4)하는 제3단계와, 상기 제3단계의 판단결과, 상기 프로세서 코어(23)가 데이터 메모리(25)의 데이터를 읽어오면, 브레이크 메모리 컨트롤 레지스터(31)의 어드레스 트레이스 체크 플래그(ACF)와 데이터 체크 플래그(DCF)를 셋(ST5)하고, 다시 현재 프로세서 코어(23)가 사용하는 데이터 메모리(25) 어드레스를 어드레스 비교부(33)에서 비교(ST7)하는 제4단계와, 상기 제4단계의 비교결과, 상기 관측대상 어드레스와 현재 프로세서 코어(23)가 사용하는 데이터 메모리(25) 어드레스가 동일하면, 프로세서 코어(23)가 데이터를 기록하는 것인지를 판단(ST9)하는 제5단계와, 상기 제5단계의 판단결과, 상기 프로세서 코어(23)가 데이터 메모리(25)에 데이터를 기록하면, 프로세서 코어(23)의 동작을 중지(ST10)시키는 제6단계로 동작 한다.

- <27>        상기 제3단계의 결과가 프로세서 코어(23)가 데이터 메모리(25)에 데이터를 기록하면 메모리 브레이크 컨트롤 레지스터(31)의 데이터 체크 플래그(DCF)를 셋(ST6)하고 프로세서 코어(23)의 동작을 중지(ST10)시킨다.
- <28>        또한, 상기 제4단계의 결과가 상기 관측대상 어드레스와 현재 프로세서 코어(23)가 사용하는 데이터 메모리(25) 어드레스가 다르면 현재 프로세서 코어(23)가 읽어온 데이터 메모리(25)의 어드레스와 데이터를 호스트 컴퓨터(20)로 출력(ST8)하고 다음 연산에서 현재 프로세서 코어(23)가 사용하는 데이터 메모리(25) 어드레스를 어드레스 비교부(33)에서 비교 한다.
- <29>        또한, 상기 제5단계의 결과가 상기 프로세서 코어(23)가 데이터를 읽어오면 다음 연산에서 현재 프로세서 코어(23)가 사용하는 데이터 메모리(25) 어드레스를 어드레스 비교부(33)에서 비교(ST7) 한다.
- <30>        이와 같이 구성 및 동작하는 본 발명을 설명 한다.
- <31>        상기 어드레스 레지스터(32)에서 하나의 데이터를 관측하는 경우는, 디버그 모드 상태에 진입하면 상기 호스트 컴퓨터(20)는 디버거 리셋에 의해 상기 메모리 브레이크 컨트롤러(26)의 메모리 브레이크 컨트롤 레지스터(31)에 MBEF=1, ACF=0, DCF=0 을 주어 초기화를 시키고 상기 데이터 메모리(25)에서 관측할 어드레스를 상기 메모리 브레이크 컨트롤러(26)의 어드레스 레지스터(32)에 저장 시킨다.

- <32>        상기 프로그램 메모리(24)에 저장된 프로그램에 따라 프로세서 코어(23)는 동작하게 되고, 상기 메모리 브레이크 컨트롤러(26)의 어드레스 비교부(33)는 상기 데이터 메모리(25)의 어드레스를 감시하게 된다.
- <33>        상기 어드레스 비교부(33)가 상기 어드레스 레지스터(32)에 저장되어 관측할 데이터 메모리(25)의 어드레스를 감지하면 상기 어드레스 비교부(33)는 상기 디버거 제어부(21)로 신호(AC=1)를 보내고 상기 디버거 제어부(21)는 상기 데이터 메모리(25)의 어드레스에 대한 데이터를 읽어오는(Read) 것인지, 아니면 상기 데이터 메모리(25)의 어드레스에 데이터를 기록하는(Write) 것인지를 판단한다.
- <34>        만약, 기록(Write) 상태라면 상기 메모리 브레이크 컨트롤 레지스터(31)는 데이터 체크 플래그를 셋하고(DCF=1) 상기 디버거 제어부(21)로 프로그램의 실행을 중지 시키는 신호를 보내게 되고 상기 프로세서 코어(23)는 중지 하게 된다.
- <35>        그러나, 만약 읽어오는(Read) 상태라면 상기 메모리 브레이크 컨트롤 레지스터(31)는 어드레스 트레이스 체크 플래그와 데이터 체크 플래그를 셋(ACF=1, DCF=1)한다. 프로세서 코어(23)가 데이터 메모리(25)의 특정 어드레스의 데이터를 읽어오는 것은 다른 어드레스의 데이터 값과 산술 및 논리적인 연산관계를 가지거나 데이터 메모리(25)의 특정 어드레스와 상관 관계를 가질수 밖에 없다. 따라서 읽어오는 상태에서는 어드레스 트레이스 체크 플래그와 데이터 체크 플래그를 셋(ACF=1, DCF=1) 한 후에, 다시 상기 메모리 브레이크 컨트롤러(26)의 어드레스 비교부(33)는 상기 데이터 메모리(25)의 어드레스를 감시하게 된다.

- <36>        상기 어드레스 비교부(33)가 상기 어드레스 레지스터(32)에 저장된 관측대상 데이터 메모리(25)의 어드레스를 감지하면 상기 어드레스 비교부(33)는 상기 디버거 제어부(21)로 신호( $AC=1$ )를 보낸다.
- <37>        만약, 상기 어드레스 비교부(33)의 값이  $AC \neq 1$  이면 관측대상 어드레스와 산술적 논리적으로 관련된 데이터 메모리(25)의 다른 어드레스 및 데이터이기 때문에 디버깅 작업을 위해 상기 메모리 브레이크 컨트롤러(26)의 어드레스 비교부(33)에 있는 어드레스와 데이터 비교부(35)에 있는 데이터를 상기 디버거 제어부(21)를 통해 호스트 컴퓨터(20)로 전송한다.
- <38>        그러나, 만약 상기 어드레스 비교부(33)의 값이  $AC=1$  이면 이전에 읽어온 관측 대상 어드레스가 사용되는 것이기 때문에 다시 상기 디버거 제어부(21)는 상기 프로세서 코어(23)가 상기 데이터 메모리(25)의 관측 대상 어드레스의 데이터를 읽어오는 (Read) 것인지, 아니면 상기 프로세서 코어(23)가 어떤 데이터 값을 관측대상 어드레스에 기록하는(Write) 것인지를 판단한다.
- <39>        만약, 읽어오는(Read) 것이면 이전에 읽어온 관측 대상 어드레스 이기 때문에 다시 어드레스를 감지하는 작업으로 돌아간다.
- <40>        만약, 기록하는(Write) 것이면 이전에 읽어온 관측 대상 어드레스에 어떤 연산에 의한 결과값을 기록하는 것이기 때문에 관측대상 어드레스의 데이터 값이 변하는 것이 되므로 상기 디버거 제어부(21)에 프로세서 코어(23)의 동작을 중단시키는 신호가 전달되고 상기 디버거 제어부(21)는 프로세서 코어(23)를 중단시키고 프로세서 코어(23)의 제어권을 상기 호스트 컴퓨터(20)로 넘기게 된다.

- <41> 한 가지 프로그램을 가정해서 동작을 설명하면,
- <42>  $A0 = a$
- <43>  $A1 = b$
- <44>  $A2 = c$
- <45>  $A2 = A0 + (A1 * A2)$  라고 프로그램이 구현되어 있다고 한다.
- <46> 이때 상기의  $A0$ ,  $A1$  및  $A2$ 는 데이터 메모리(25)의 어드레스이다. 우리는 디버깅을 하기 위해서 상기 데이터 메모리(25)의 어드레스에서  $A2$ 번지가 관측대상의 어드레스라고 하자.
- <47> 상기의 프로그램을 디버그 모드에서 실행시키면 상기의 프로그램은 관측대상 어드레스인  $A2$ 와 현재 사용중인 어드레스를 비교하면서 실행되다가 3번째 줄에서  $A2$ 에 해당하는 데이터값  $c$ 를 읽기(Read) 때문에 관측대상 어드레스인  $A2$ 를 어드레스 비교부(33)가 감지하게 될 것이며 프로세서 코어(23)는 멈추게 된다. 그 결과로서  $A2$ 의 어드레스와 데이터값  $c$ 를 상기 호스트 컴퓨터(20)에 전달하게 된다.
- <48> 그리고, 계속해서 실행 시킬 경우 4번째 줄에서 다시 멈추게 될 것이며 우변의  $A0$ ,  $A1$  및  $A2$ 를 관측대상 어드레스인  $A2$ 와 비교하게 될 것이다.  $A0$  및  $A1$ 과 관측대상 어드레스를 비교한 후에는  $A0$  및  $A1$ 에 해당하는 어드레스와 데이터값을 호스트 컴퓨터(20)로 전송하고,  $A2$ 와 비교한 후에는 다시 읽기(Read)인지 기록(Write)인지를 판단하게 되는데 여기서는 읽기(Read)이므로 상기에서 확인된 내용이므로 그냥 지나가게 된다. 상기 4번째 줄의 우변을 비교한 후에는 다시 좌변

을 비교하게 되는데 A2를 감지하고 다시 읽기(Read)인지 기록(Write)인지를 판단하게 되는데 여기서는 계산 결과를 기록(Write)하는 것이므로 프로세서 코어(23)는 멈추게 된다.

<49>       상기 어드레스 레지스터(32)에서 배열로 구성된 데이터를 관측하는 경우는, 상기 동작 설명에 있어서 상기 메모리 브레이크 컨트롤러(26)의 어드레스 레지스터(32)에 배열의 최상위 어드레스와 최하위 어드레스를 저장하고, 상기 데이터 레지스터(34)에 배열의 데이터를 저장 할수 있는 공간을 설정한 후, 상기 어드레스 비교부(33)에서 상기 어드레스 레지스터(32)에 저장된 배열의 범위와 비교하여 배열의 범위에 해당하는 상기 데이터 메모리(25)의 어드레스를 감지 한다. 이후의 동작은 상기 하나의 데이터를 관측하는 경우와 동일한 동작을 수행하게 된다.

#### 【발명의 효과】

<50>       이상에서 상세히 설명한 바와 같이 본 발명은 프로세서의 디버깅 작업에 있어서 데이터 메모리의 특정 어드레스의 데이터를 감시하여 데이터의 흐름 및 변화를 감지 하기때문에 디버깅 작업시 시간과 비용을 절약 할수 있는 효과가 있다.



**【특허청구범위】****【청구항 1】**

프로그램 메모리에 저장된 프로그램에 따라 동작하는 프로세서 코어와; 디버깅을 수행하는 호스트 컴퓨터로부터 명령을 받아 상기 프로세서 코어를 제어하는 디버거 제어부와; 상기 디버거 제어부를 통하여 상기 호스트 컴퓨터로부터 브레이크 포인트 어드레스를 받아 상기 프로세서 코어가 사용하는 프로그램 메모리의 어드레스를 관측하고, 동일한 어드레스가 감지되면 디버거 브레이크 포인트로 인식하여 상기 디버거 제어부로 신호를 전달하는 브레이크 포인트 감지부와; 상기 프로세서 코어의 데이터를 저장하는 데이터 메모리와; 상기 데이터 메모리의 어드레스와 데이터를 감시하고 해당 어드레스를 감지시 상기 디버거 제어부로 신호를 보내어 프로세서 코어의 동작을 중지시키고 해당 어드레스와 데이터를 출력하는 메모리 브레이크 컨트롤러로 구성된 것을 특징으로 하는 데이터의 천이 상태에 따른 디버깅 장치.

**【청구항 2】**

제1항에 있어서, 상기 메모리 브레이크 컨트롤러는 상기 디버거 제어부의 신호에 의해 활성화 되는 메모리 브레이크 컨트롤 레지스터와; 상기 메모리 브레이크 컨트롤 레지스터로부터 관측대상 어드레스를 받아 저장하는 어드레스 레지스터와; 상기 데이터 메모리로부터 받은 현재 프로세서 코어가 사용하는 데이터 메모리의 어드레스와 상기 어드레스 레지스터에 저장된 관측대상 어드레스를 비교하는 어드레스 비교부와; 상기 어드레스 레지스터에 저장되어 있는 브레이크 포인트 어드레스의 데이터 값을 저장하는 데이터 레지스터와; 상기 데이터 메모리

리로 부터 받은 현재 데이터와 상기 데이터 레지스터에 저장되어있는 관측대상 어드레스의 데이터 값을 비교하는 데이터 비교부로 구성된 것을 특징으로 하는 데이터 천이 상태에 따른 디버깅 장치.

### 【청구항 3】

디버거 모드로 전환시, 관측대상 어드레스를 저장하고 현재 프로세서 코어가 사용하는 데이터 메모리 어드레스와 상기 관측대상 어드레스를 비교하는 제1단계와; 상기 제1단계의 비교결과, 어드레스가 동일하면, 프로세서 코어가 상기 관측대상 어드레스에 해당하는 데이터를 읽어오는 것인지 기록하는 것인지를 판단하는 제2단계와; 상기 제2단계의 판단결과, 프로세서 코어가 데이터를 읽어오면, 어드레스 트레이스 체크 플래그와 데이터 체크 플래그를 셋하고 프로세서 코어의 동작을 중지시키는 제3단계와; 상기 제2단계의 판단결과, 프로세서 코어가 데이터를 기록하면, 데이터 체크 플래그를 셋하고 프로세서 코어의 동작을 중지시키는 제4단계로 동작하는 것을 특징으로 하는 데이터 천이 상태에 따른 디버깅 방법.

### 【청구항 4】

제3항에 있어서, 제3단계에서 어드레스 트레이스 체크 플래그와 데이터 체크 플래그를 셋 한후, 다시 현재 프로세서 코어가 사용하는 데이터 메모리 어드레스와 관측대상 어드레스를 비교하여 동일하지 않은경우, 현재 어드레스 레지스터와 데이터 레지스터에 저장된 어드레스와 데이터를 출력시키고 다시 프로세서 코어가 사용하는 다음 데이터 메모리 어드레스와 관측대상 어드레스를 비교는 것을 특징으로 하는 데이터 천이 상태에 따른 디버깅 방법.

**【청구항 5】**

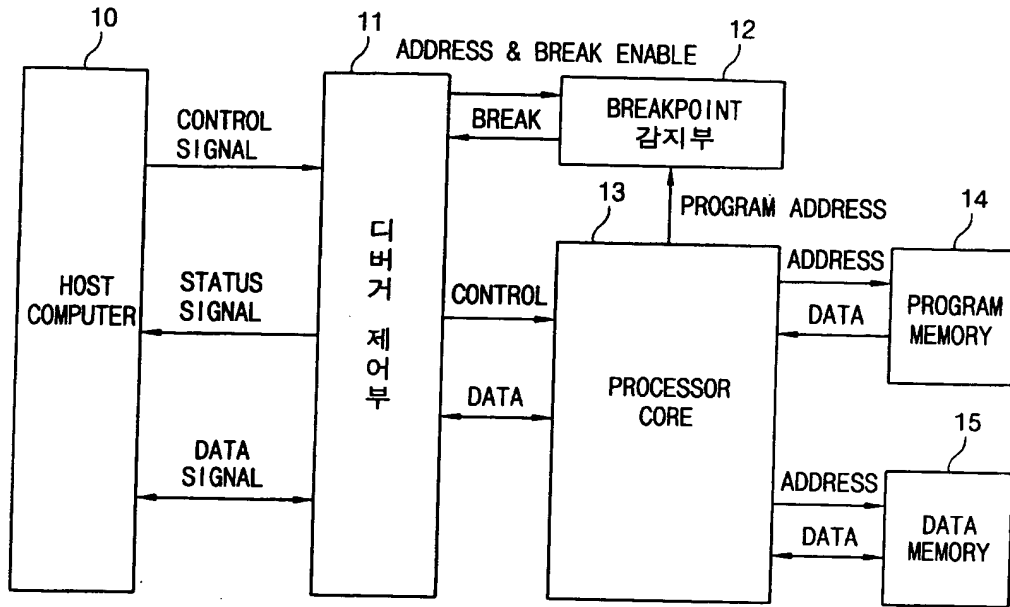
제3항에 있어서, 제3단계에서 어드레스 트레이스 체크 플래그와 데이터 체크 플래그를 셋 한후, 다시 현재 프로세서 코어가 사용하는 데이터 메모리 어드레스와 관측대상 어드레스를 비교하여 동일한 경우, 읽기인지 기록인지를 판단하여 기록이면 프로세서 코어의 동작을 중지시키는 것을 특징으로 하는 데이터 천이 상태에 따른 디버깅 방법.

**【청구항 6】**

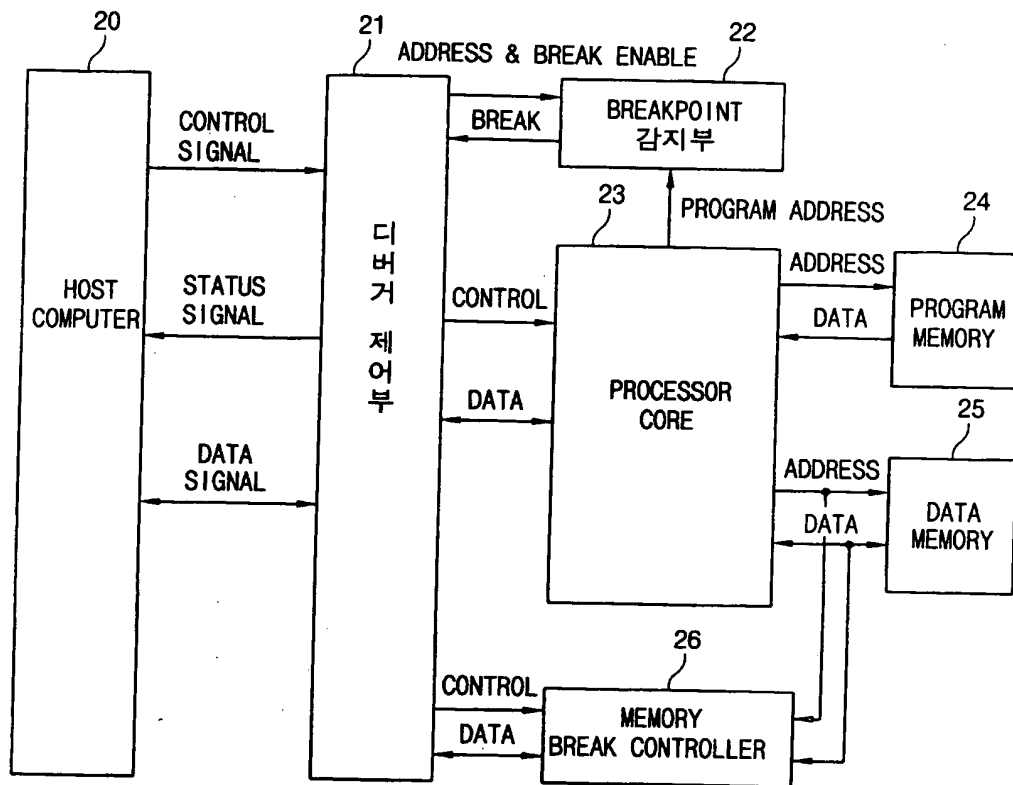
제5항에 있어서, 읽기이면 다시 프로세서 코어가 사용하는 다음 데이터 메모리 어드레스와 관측대상 어드레스를 비교는 것을 특징으로 하는 데이터 천이 상태에 따른 디버깅 방법.

【도면】

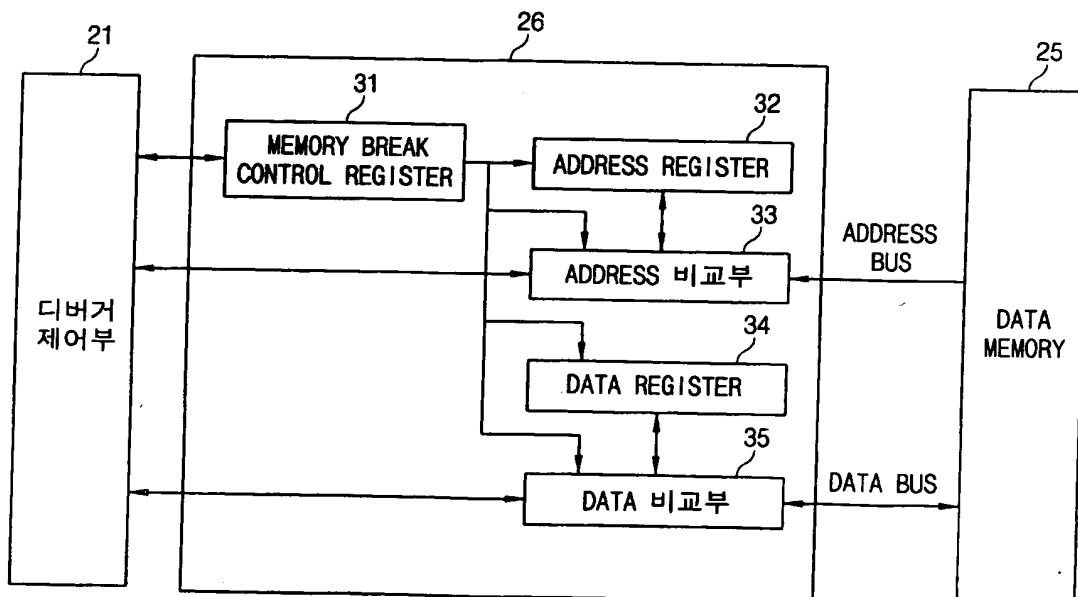
【도 1】



【도 2】



【도 3】



【도 4】

